IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re the Application of:	)	Group Art Unit: 2192
DAVDA, Bhavesh P.	)	Examiner: Wei, Zheng
Serial No.: 10/650,257	)	Confirmation No.: 7452
Filed: August 27, 2003	)	REASONS SUPPORTING PRE-APPEAL BRIEF REQUEST FOR REVIEW
Atty. File No.: 4366-120	)	
For: "METHOD AND APPARATUS FOR HOT UPDATING OF RUNNING PROCESSES"	)	CERTIFICATE OF TRANSMISSION I HEREBY CERTIFY THAT THIS CORRESPONDENCE IS BEING TRANSMITTED VIA EFS-WEB PURSUANT TO §1.6(a)(4) ON October 8, 2007. SHERIDAN ROSS P.C. BY: Margaret Howard

Mail Stop AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313

Dear Sir:

The following sets forth Applicant's reasons in support of the Pre-Appeal Brief Request for Review submitted herewith.

The Examiner's rejections omit essential elements needed to reject the pending claims. In particular, the cited references do not teach, suggest or describe a system in which the first instruction of a replaced function is replaced by a jump instruction and an address of an update table comprising an address of a first instruction of a first updated function as recited by independent Claims 1 and 10 or updating an executing program as recited by independent Claims 18 and 20. Accordingly, it is submitted that all of the claims are in condition for allowance.

The Examiner rejects Claims 1-5, 7, 10-13 and 16-19 under 35 U.S.C. §102 as being anticipated by U.S. Patent No. 5,619,698 to Lillich et al. ("Lillich"). In addition, Claims 6, 8, 9, 14, 15, 20, and 21 stand rejected under 35 U.S.C. § 103 as being obvious over Lillich in view of U.S. Patent Application Publication No. 2004/0107416 to Buban et al. ("Buban"), Lillich in view of U.S. Patent Application Publication No. 2003/0167463 to Munsil et al. ("Munsil"), or U.S. Patent Application Publication No. 2002/0152455 to Hundt et al ("Hundt") in view of Buban. However, none of the cited references discloses injecting a jump instruction and an

address of an update table at a location in memory or updating an executing program as claimed. Accordingly, the rejections of the claims should be reconsidered and withdrawn.

The Lillich reference is generally directed to a method and apparatus for patching operating systems. In the background discussion provided by Lillich, which has been cited by the Office Action in connection with the final rejection of the claims, a client call made to a replaced function is redirected to a patch comprising the computer code replacing the called function. In this conventional patch arrangement, an ATRAP instruction is included in application code. In response to reaching the ATRAP instruction, the microprocessor running the code will push the current state on to the computer's stack, and resume execution at the address indicated in a low memory location. (Lillich, col. 2, lines 48-54.) The indicated address is the address of the TRAP dispatcher. The TRAP dispatcher then operates to determine the operation indicated by the ATRAP instruction, looks up the address of the corresponding system routine in the TRAP table, and then jumps to the corresponding system routine. (Lillich, col. 3, lines 1-5.)

Accordingly, it can be appreciated that Lillich does not teach, suggest or describe injecting a jump instruction and an address of an update table at a location in a memory containing a first instruction of a first replaced function as recited by independent Claims 1 and 10. In addition, Lillich does not teach, suggest or describe stopping execution of program code and inserting a jump code and an address associated with a replacement function in place of an address of the function to be replaced in existing executable program code as recited by Claim 18.

The Buban and Munsil references that are cited by the final Office Action in connection with rejections of various dependent claims do not teach, suggest or describe those elements of the independent claims that are absent from Lillich. In addition, Buban does not discuss a predetermined distance between memory addresses and populating an update table with the address of the first updated function in response to determining that the first distance is at least the predetermined amount. Instead, Buban discusses preferably limiting the size of the instruction being updated to the processor's smallest unit of automatically replaceable memory. (Buban, paragraph 46.) Therefore, the Lillich and the Buban references do not teach, suggest or disclose the elements of Claims 6, 8, 14 and 15.

The Munsil reference is cited in connection with a second memory space that is read-only

memory space. However, Munsil does not teach, suggest or describe elements of the independent claims that are not disclosed by the other cited references and Claim 9 is therefore not obvious.

The Office Action rejects Claims 20 and 21 as being obvious over Hundt in view of Buban. Claim 20 recites a system for updating executing program code that includes a signal handler tool operable to replace in memory an address of the function to be replaced with the address of the replacement function. The replacement is performed in response to the instruction pointer being at least a predetermined distance from the address for the replacement function. Claim 21 depends from Claim 20.

The Hundt reference discusses using original functions to generate instrumented functions in executing the instrumented functions in place of the original functions. (Hundt, paragraph 18.) The Office Action acknowledges that there is no disclosure in Hundt related to making such a replacement in response to the position of the instruction pointer being at least a predetermined distance from the address of the replacement function. For such a disclosure, the Office Action cites Buban. However, Buban in fact contains no teaching, suggestion or disclosure of such a pre-determined distance. Therefore the rejections of Claims 20 and 21 should be reconsidered and withdrawn.

Accordingly, at least the portions of the independent claims indicated in italicized text, as set forth below, cannot be found in the cited references:

1. (Currently Amended) A method for updating a running process, comprising:

allocating in executable program code text first memory space operable to receive new program instructions comprising at least a first updated function;

allocating in executable program code text second memory space operable to receive address information related to said new program instructions;

running said executable program code;

stopping execution of said executable program code;

injecting a jump instruction and an address of an update table at a location in a memory containing a first instruction of a first replaced function, wherein said address of said update table contains an address of a first instruction of said first updated function; and

resuming execution of said executable program code, wherein said first updated function

is called in place of said first function, and wherein said executable code is updated in said memory.

10. (Previously Presented) A computer implemented method, the method comprising:

receiving information identifying:

a running executable program to be patched; and

a function to be replaced;

accessing a symbol table in a memory for said executable program to be patched;

obtaining from said symbol table an address of said function to be replaced;

stopping execution by a processor of said running executable program to be patched;

injecting in said running executable program to be patched at a location in said memory containing a first instruction of said function to be replaced a jump instruction and an address of a new function, wherein said new function is executed by said processor in place of said function to be replaced, and wherein a patched version of said executable program is created in said memory; and

resuming execution of said executable program by said processor, wherein said patched version of said executable program is executed by said processor.

18. (Original) A system for updating executable program code, comprising:

means for receiving information identifying existing executable program code;

means for receiving information identifying a function to be replaced;

means for stopping execution of said existing executable program code;

means for inserting a jump code and an address associated with a replacement function in place of an address of said function to be replaced in said existing executable program code, wherein updated executable program code is created; and

means for resuming execution of said executable program code, wherein said updated executable program code is executed.

20. (Original) A system *for updating executing program code*, comprising: a create patch tool operable to receive information identifying an executable program to be updated and a function to be replaced;

a patch tool operable to query an operating system for a process identifier associated with said identified executable program;

a debugging utility operable to stop execution of said executable program to be updated and to determine a position of an instruction pointer associated with said executable program to be updated; and

a signal handler tool operable to replace in memory an address of said function to be replaced with an address of a replacement function in response to said position of said instruction pointer being at least a predetermined distance from said address of said replacement function, wherein said replacement function is executed instead of said function to be replaced upon resuming execution of said executable program, wherein said executable program is updated.

Because the references cited in the final Office Action do not teach, suggest or disclose a system or method as claimed, and in particular do not teach, suggest or describe injecting a jump instruction in an address of an update table in a location of memory containing a first instruction of a first replaced function as recited by Claims 1 and 10, or updating an executing program as cited by Claims 18 and 20, essential elements required for a rejection of the claims have been omitted by the final Office Action. Therefore, the rejections of the claims in view of the cited references should be reconsidered and withdrawn, and the claims allowed.

The Pre-Appeal Brief Conference participants are invited to contact the undersigned by telephone if doing so would be of assistance.

Respectfully submitted,

SHERIDAN ROSS P.C.

By: _____

Bradley M. Knepper
Registration No. 44,189
1560 Broadway, Suite 1200
Denver, Colorado 80202-5141
(303) 863-9700

Date: October 8, 2007

Doc Code: AP.PRE.REQ

PTO/SB/33 (07-05)
Approved for use through xx/xx/200x. OMB 0651-00xx
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PRE-APPEAL BRIEF REQUEST FOR REVIEW

Docket Number (Optional): 4366-120

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to "Mail Stop AF, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450" [37 CFR 1.8(a)]

on _____

Signature _____

Typed or printed name _____

Application Number	Filed
10/650,257	August 27, 2003

First Named Inventor: DAVDA, Bhavesh P.

Art Unit	Examiner
2192	WEI, Zheng

Applicant requests review of the final rejection in the above-identified application. No amendments are being filed with this request.

This request is being filed with a notice of appeal.

The review is requested for the reason(s) stated on the attached sheet(s).
Note: No more than five (5) pages may be provided.

I am the

- [ ] applicant/inventor.
- [ ] assignee of record of the entire interest. See 37 CFR 3.71. Statement under 37 CFR 3.73(b) is enclosed. (Form PTO/SB/96)
- [X] attorney or agent of record. Registration number 44,189
- [ ] attorney or agent acting under 37 CFR 1.34. Registration number if acting under 37 CFR 1.34 _____

/Bradley M. Knepper/ Signature

Bradley M. Knepper
Typed or printed name

303-863-9700
Telephone number

October 8, 2007
Date

NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required. Submit multiple forms if more than one signature is required, see below*.

- [ ] *Total of _____ forms are submitted.